

9/9

0800 Antan started
 1000 stopped - antan ✓
 1300 (032) MP - MC { 1.2700 9.037 847 025
 2.130476415 9.037 846 995 correct
 (033) PRO 2 2.130476415
 correct 2.130676415

Depuração e o Algoritmo da Banheira

1100 Started Cosine (check)
 1525 Started Multiplier test.

1545



Marco Dimas Gubitoso

Relay #70 Panel F
 (nothin relay.)

First actual case of bug being found.
 1630 antan started.
 1700 closed down.

Apresentação

- Ciclo de Depuração
 - Testes
 - Estabilização
 - Localização
 - Correção
- Alguns bugs famosos
- O Algoritmo da Banheira para Curvas de Nível (J. Plauger)

First actual case of bug being found.
1030 antanuk started.
1700 closed down.

1000 stopped - antanuk ✓
1300 (032) MP - MC 1.98264000 9.037 846 995 correct
2.130476415 (03) 4.615925059 (-2)
(032) PRO 2 2.130476415
correct 2.130676415
Relays 6-2 in 033 failed special speed test
10.000 test.

1100 Relays changed
1525 Started Cosine Tape (Sine check)
Multi-Adder Test.

1545 Relay #70 Panel F
(moth) in relay.

Relay #70
2145
Relay #70

Depuração



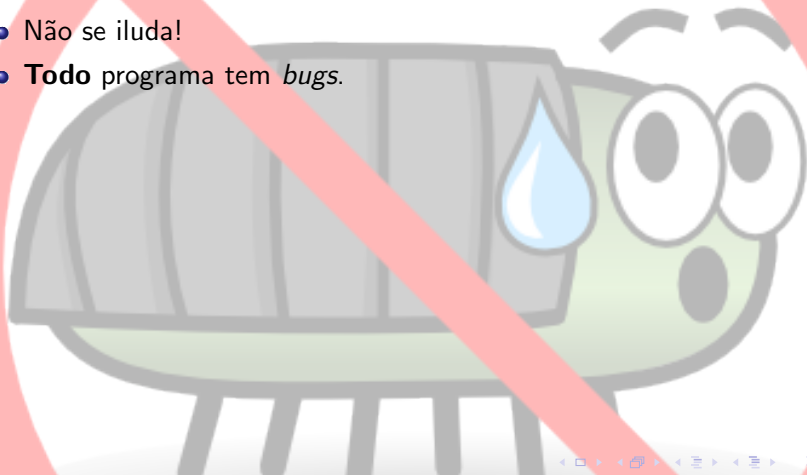
Depuração

- Não se iluda!



Depuração

- Não se iluda!
- **Todo** programa tem *bugs*.



Depuração

- Não se iluda!
- **Todo** programa tem *bugs*.
- Isto é uma consequência imediata da Lei de Murphy.

Depuração

- Não se iluda!
- **Todo** programa tem *bugs*.
- Isto é uma consequência imediata da Lei de Murphy.
- Mesmo que você tenha certeza que está tudo correto, aja como se não estivesse.

Depuração

- Não se iluda!
- **Todo** programa tem *bugs*.
- Isto é uma consequência imediata da Lei de Murphy.
- Mesmo que você tenha certeza que está tudo correto, aja como se não estivesse.
- Tenha em mente que o programa em geral funciona nos casos comuns e quebra nos casos particulares

1. Testes



1. Testes

- É a parte mais importante.

June 7th to

June 30th

2 Sims

Lemania's &



1. Testes

- É a parte mais importante.
- Uma dica é escrever os testes antes do programa:

1. Testes

- É a parte mais importante.
- Uma dica é escrever os testes antes do programa:
- Assim você evita direcionar o teste para o código escrito. É mais fácil achar situações inesperadas pelo programa.

1. Testes

- É a parte mais importante.
- Uma dica é escrever os testes antes do programa:
- Assim você evita direcionar o teste para o código escrito. É mais fácil achar situações inesperadas pelo programa.
- Peça para alguma outra pessoa usar o seu programa, mas evite ficar dando informações adicionais.

1. Testes

- É a parte mais importante.
- Uma dica é escrever os testes antes do programa:
- Assim você evita direcionar o teste para o código escrito. É mais fácil achar situações inesperadas pelo programa.
- Peça para alguma outra pessoa usar o seu programa, mas evite ficar dando informações adicionais.
- É claro que você pode fazer testes direcionados ao código.

Desenvolvimento incremental



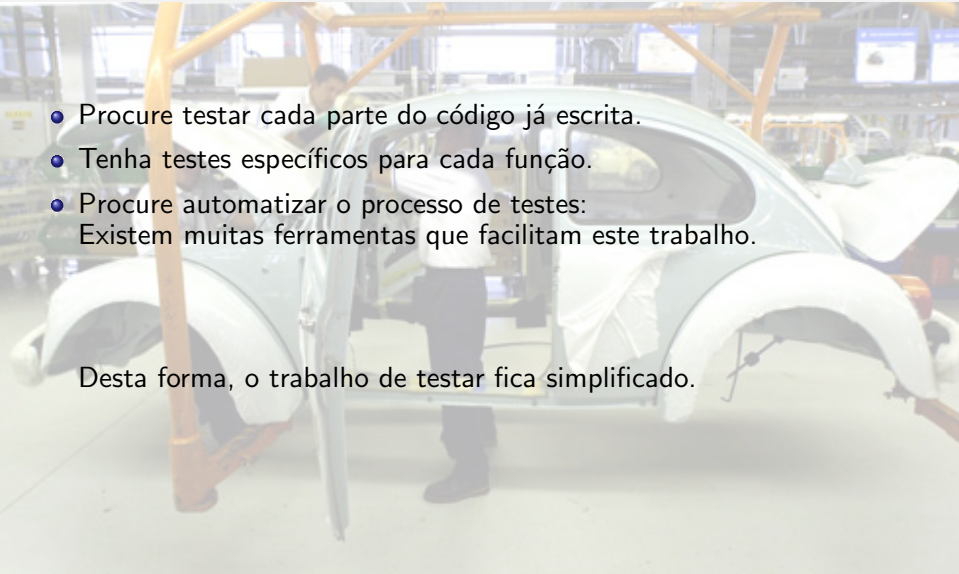
Desenvolvimento incremental

- Procure testar cada parte do código já escrita.

Desenvolvimento incremental

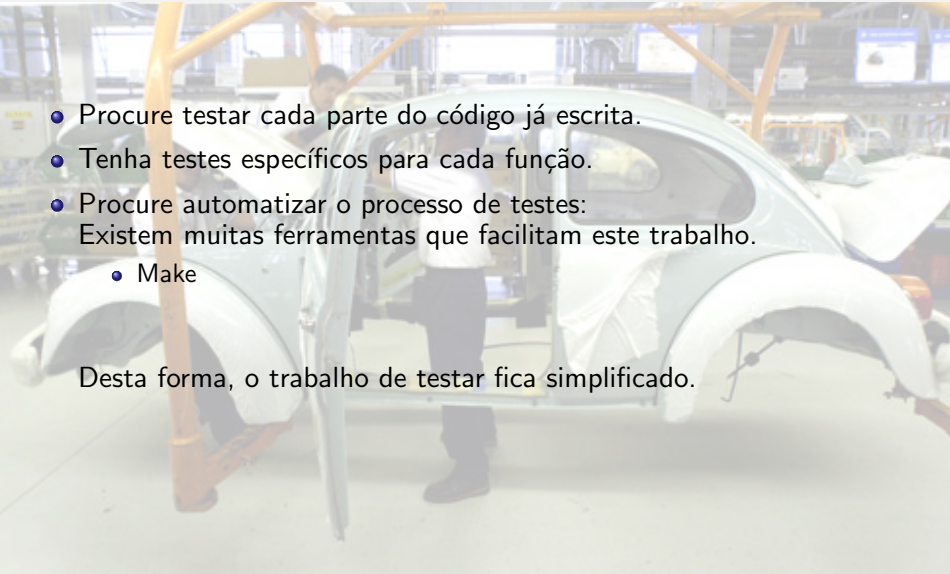
- Procure testar cada parte do código já escrita.
- Tenha testes específicos para cada função.

Desenvolvimento incremental

- 
- A light blue Volkswagen Beetle is shown on an assembly line. A worker in a white shirt and dark pants is standing next to the car, which has its driver-side door open. The car is positioned under a yellow overhead crane system. The background shows a typical factory setting with various equipment and structural elements.
- Procure testar cada parte do código já escrita.
 - Tenha testes específicos para cada função.
 - Procure automatizar o processo de testes:
Existem muitas ferramentas que facilitam este trabalho.

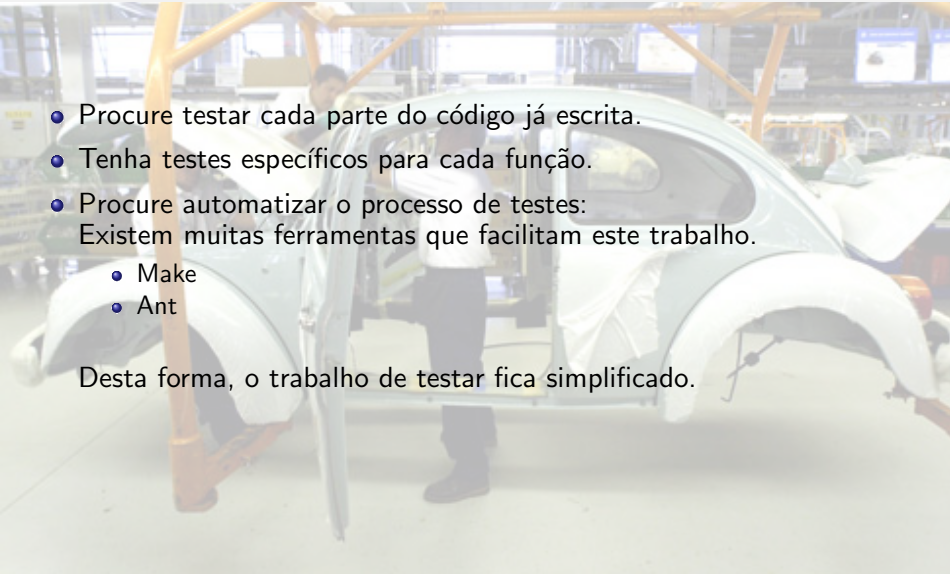
Desta forma, o trabalho de testar fica simplificado.

Desenvolvimento incremental

- 
- Procure testar cada parte do código já escrita.
 - Tenha testes específicos para cada função.
 - Procure automatizar o processo de testes:
Existem muitas ferramentas que facilitam este trabalho.
 - Make

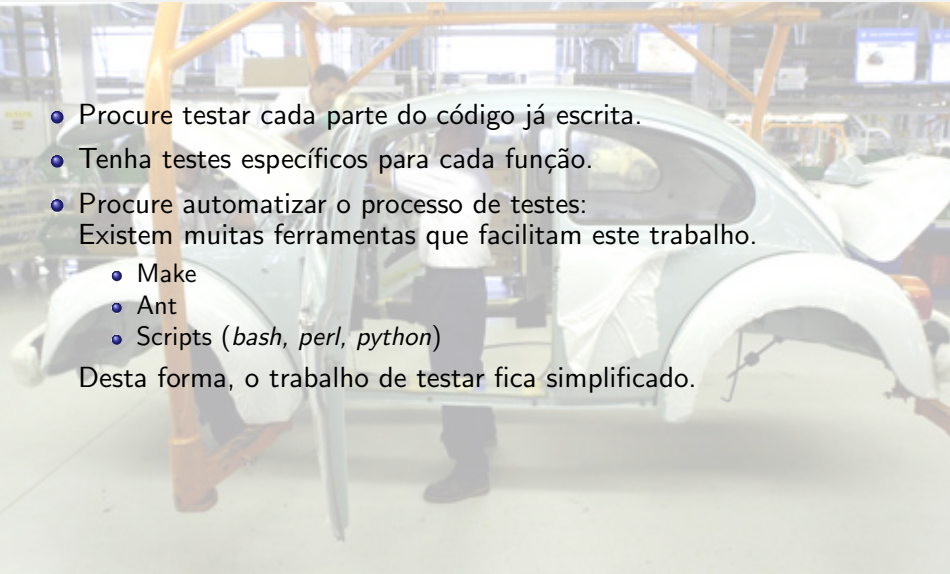
Desta forma, o trabalho de testar fica simplificado.

Desenvolvimento incremental

- 
- Procure testar cada parte do código já escrita.
 - Tenha testes específicos para cada função.
 - Procure automatizar o processo de testes:
Existem muitas ferramentas que facilitam este trabalho.
 - Make
 - Ant

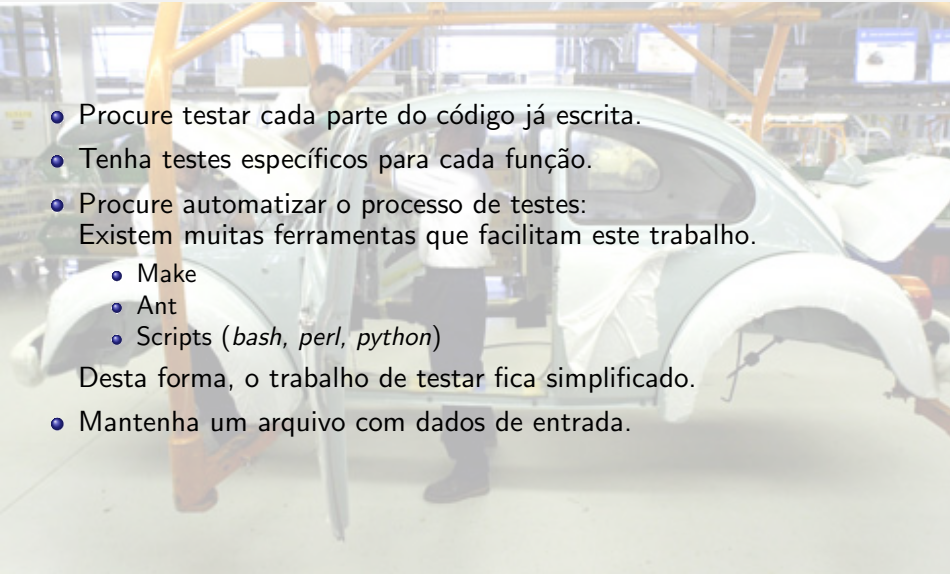
Desta forma, o trabalho de testar fica simplificado.

Desenvolvimento incremental

- 
- Procure testar cada parte do código já escrita.
 - Tenha testes específicos para cada função.
 - Procure automatizar o processo de testes:
Existem muitas ferramentas que facilitam este trabalho.
 - Make
 - Ant
 - Scripts (*bash*, *perl*, *python*)

Desta forma, o trabalho de testar fica simplificado.

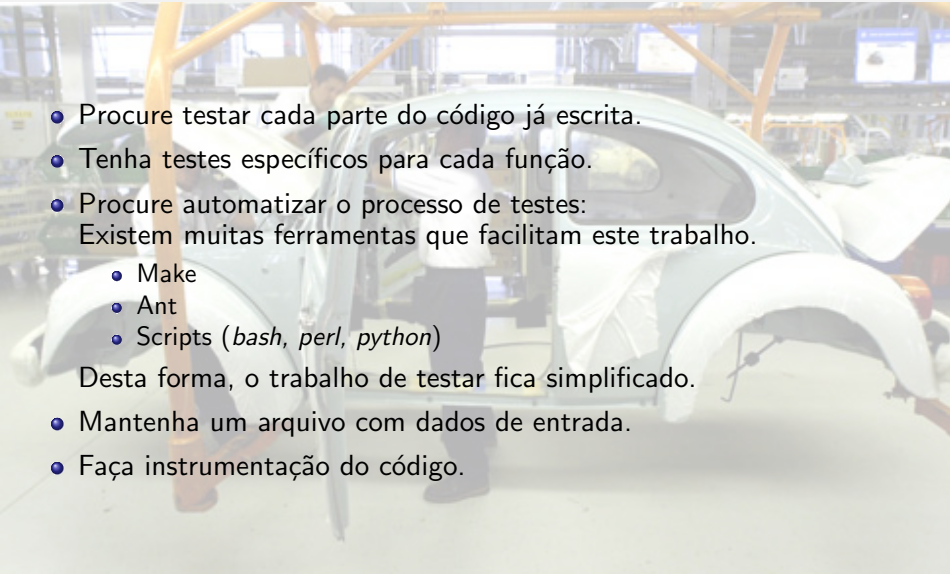
Desenvolvimento incremental

- 
- Procure testar cada parte do código já escrita.
 - Tenha testes específicos para cada função.
 - Procure automatizar o processo de testes:
Existem muitas ferramentas que facilitam este trabalho.
 - Make
 - Ant
 - Scripts (*bash, perl, python*)

Desta forma, o trabalho de testar fica simplificado.

- Mantenha um arquivo com dados de entrada.

Desenvolvimento incremental

- 
- Procure testar cada parte do código já escrita.
 - Tenha testes específicos para cada função.
 - Procure automatizar o processo de testes:
Existem muitas ferramentas que facilitam este trabalho.
 - Make
 - Ant
 - Scripts (*bash*, *perl*, *python*)

Desta forma, o trabalho de testar fica simplificado.

- Mantenha um arquivo com dados de entrada.
- Faça instrumentação do código.

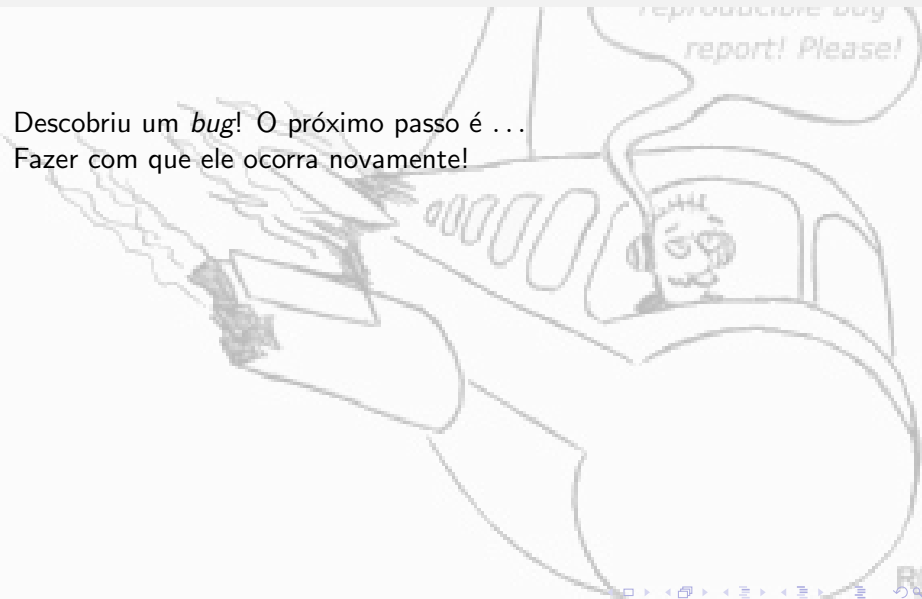
2. Estabilização

Descobriu um *bug*! O próximo passo é ...



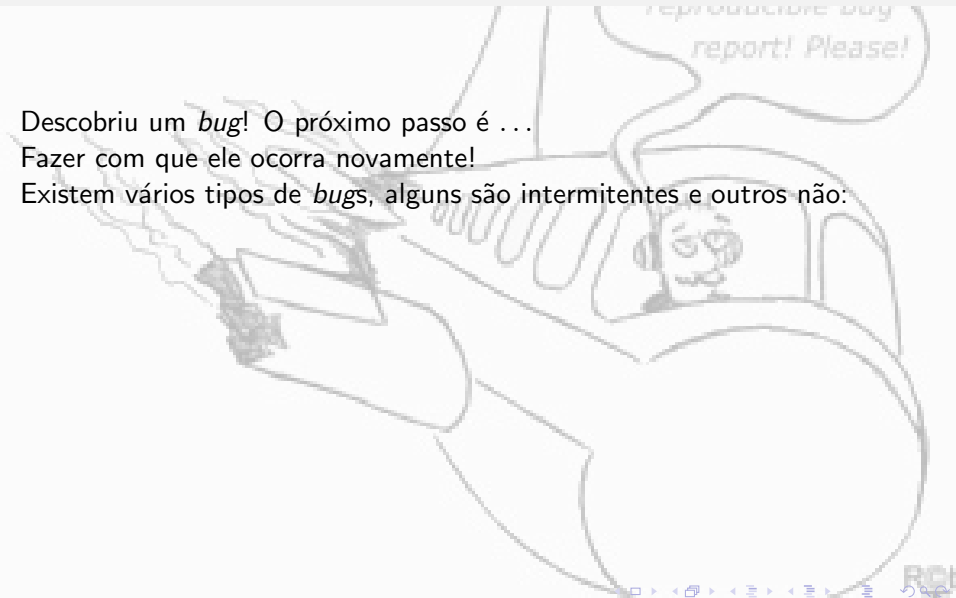
2. Estabilização

Descobriu um *bug*! O próximo passo é ...
Fazer com que ele ocorra novamente!



2. Estabilização

Descobriu um *bug*! O próximo passo é ...
Fazer com que ele ocorra novamente!
Existem vários tipos de *bugs*, alguns são intermitentes e outros não:



2. Estabilização

Descobriu um *bug*! O próximo passo é ...

Fazer com que ele ocorra novamente!

Existem vários tipos de *bugs*, alguns são intermitentes e outros não:

- Crash



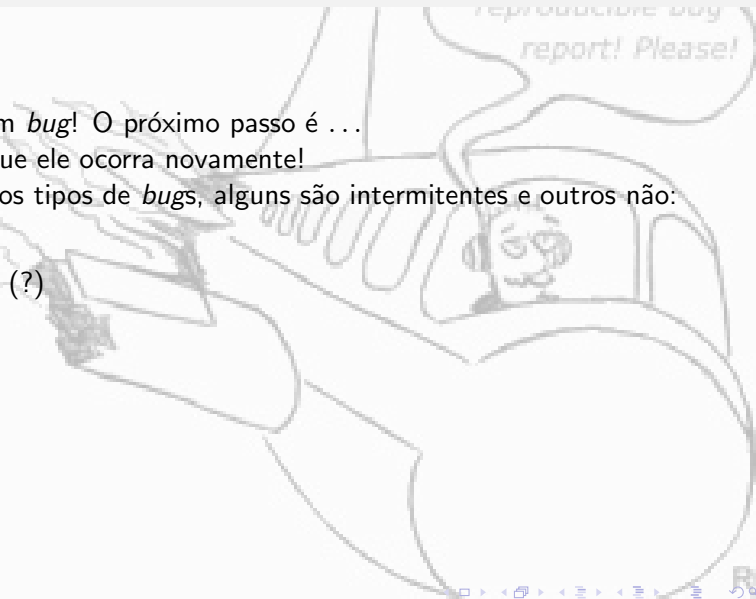
2. Estabilização

Descobriu um *bug*! O próximo passo é ...

Fazer com que ele ocorra novamente!

Existem vários tipos de *bugs*, alguns são intermitentes e outros não:

- Crash
- Usuário (?)



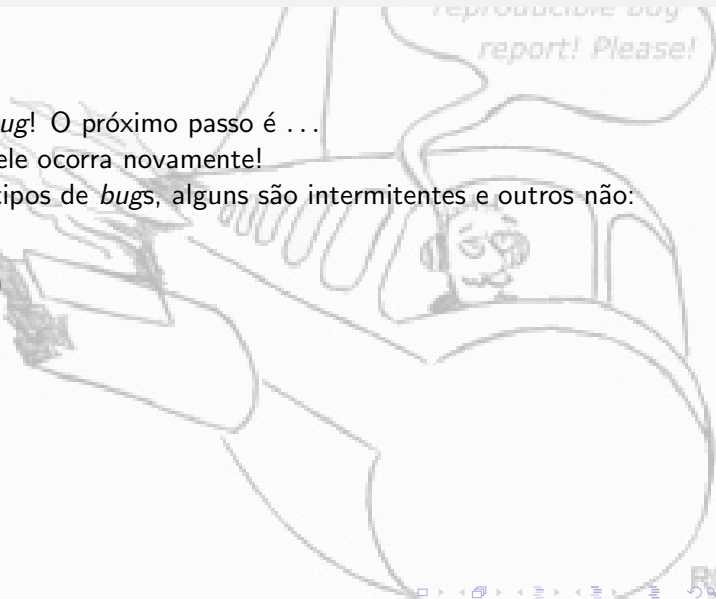
2. Estabilização

Descobriu um *bug*! O próximo passo é ...

Fazer com que ele ocorra novamente!

Existem vários tipos de *bugs*, alguns são intermitentes e outros não:

- Crash
- Usuário (?)
- Heisenberg



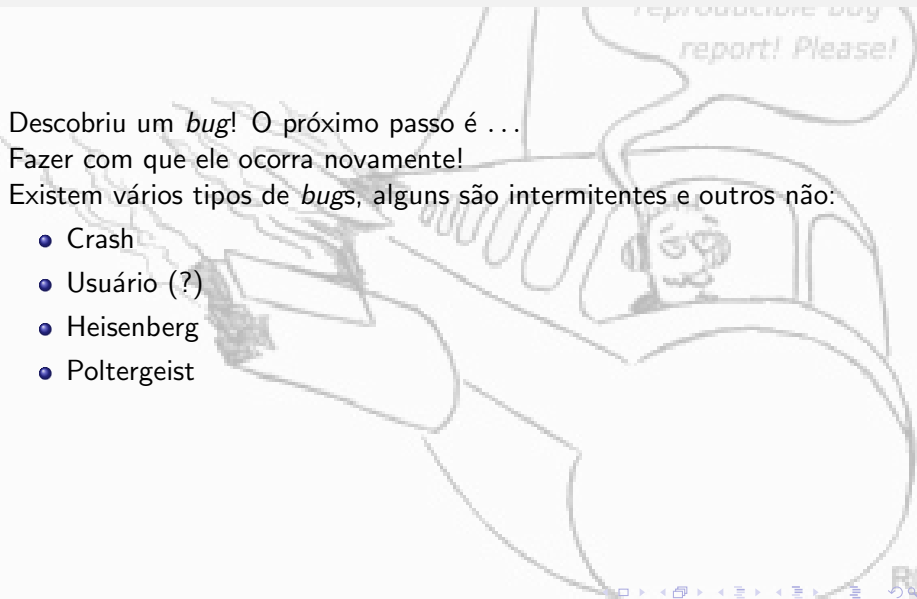
2. Estabilização

Descobriu um *bug*! O próximo passo é ...

Fazer com que ele ocorra novamente!

Existem vários tipos de *bugs*, alguns são intermitentes e outros não:

- Crash
- Usuário (?)
- Heisenberg
- Poltergeist



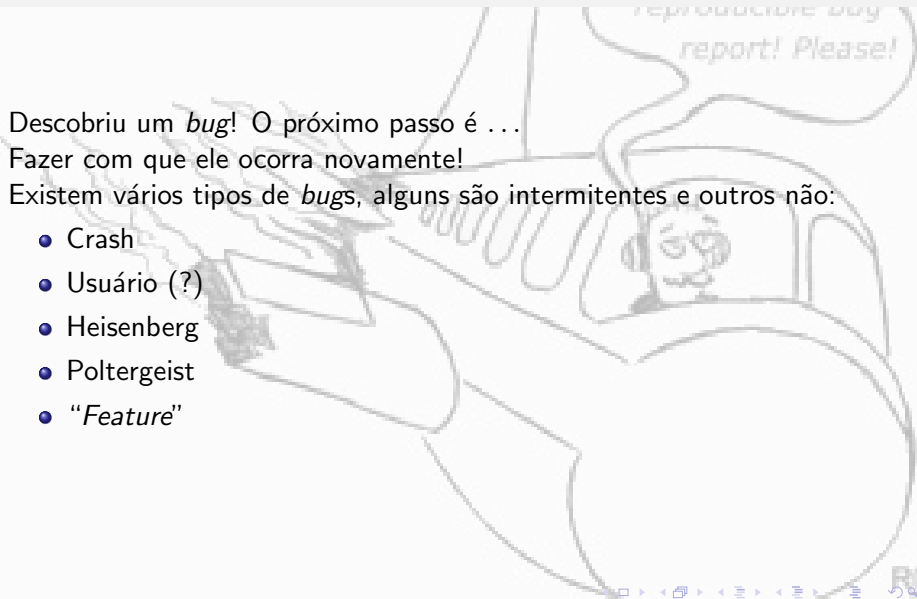
2. Estabilização

Descobriu um *bug*! O próximo passo é ...

Fazer com que ele ocorra novamente!

Existem vários tipos de *bugs*, alguns são intermitentes e outros não:

- Crash
- Usuário (?)
- Heisenberg
- Poltergeist
- “*Feature*”



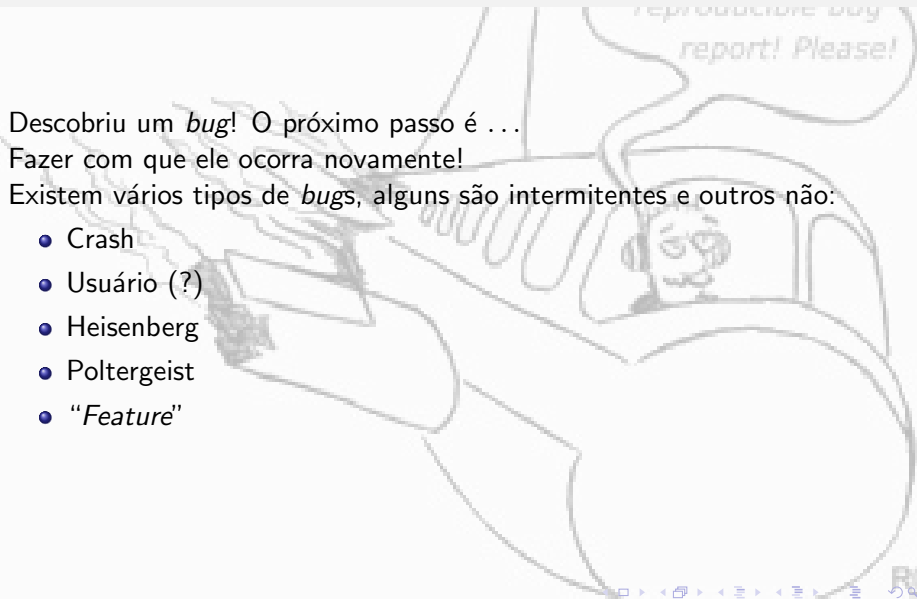
2. Estabilização

Descobriu um *bug*! O próximo passo é ...

Fazer com que ele ocorra novamente!

Existem vários tipos de *bugs*, alguns são intermitentes e outros não:

- Crash
- Usuário (?)
- Heisenberg
- Poltergeist
- “*Feature*”



2. Estabilização

Descobriu um *bug*! O próximo passo é ...

Fazer com que ele ocorra novamente!

Existem vários tipos de *bugs*, alguns são intermitentes e outros não:

- Crash
- Usuário (?)
- Heisenberg
- Poltergeist
- “*Feature*”

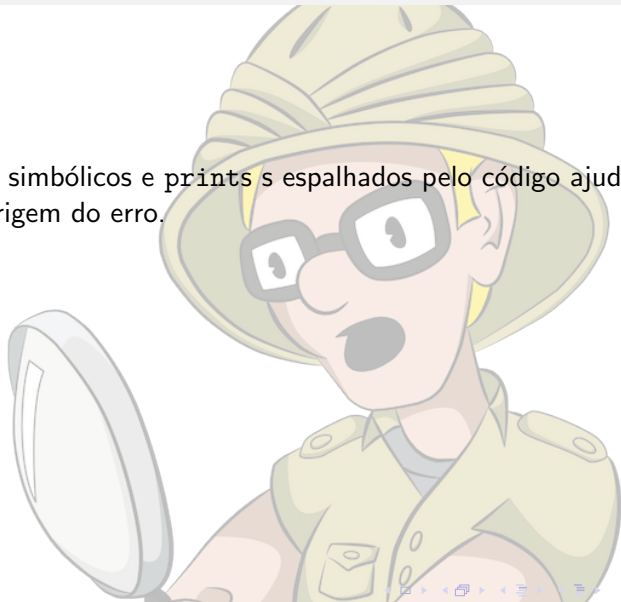
É muito importante ser capaz de gerar o erro de forma controlada.

3. Localização



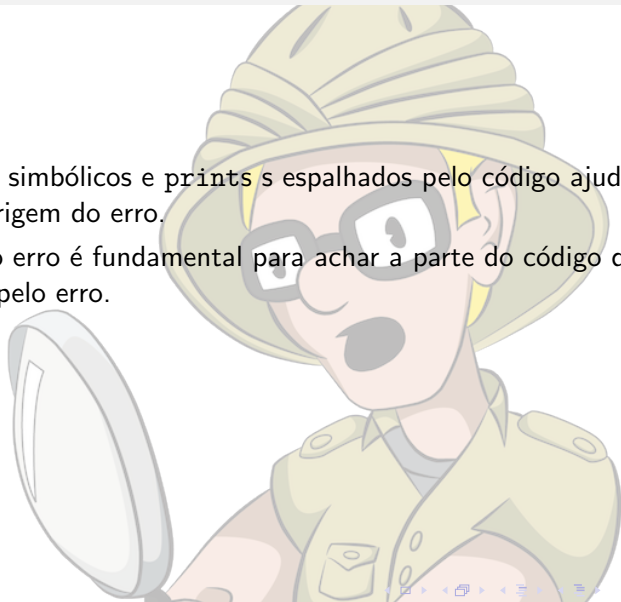
3. Localização

- Depuradores simbólicos e `prints` espalhados pelo código ajudam a localizar a origem do erro.



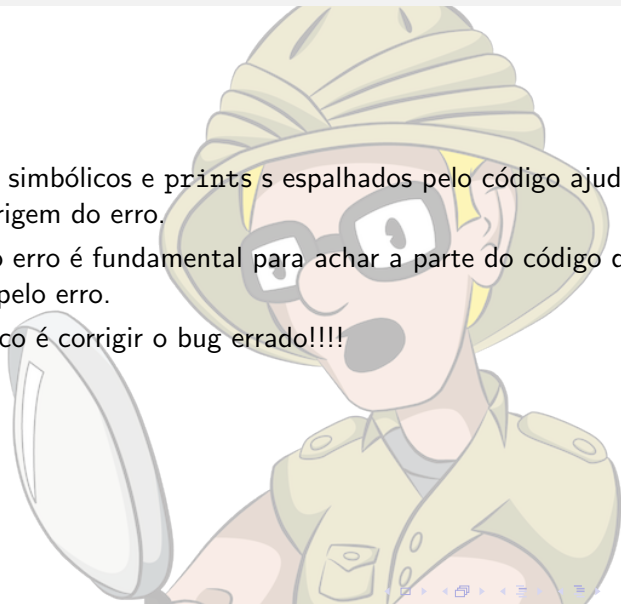
3. Localização

- Depuradores simbólicos e `prints` espalhados pelo código ajudam a localizar a origem do erro.
- Reproduzir o erro é fundamental para achar a parte do código que é responsável pelo erro.



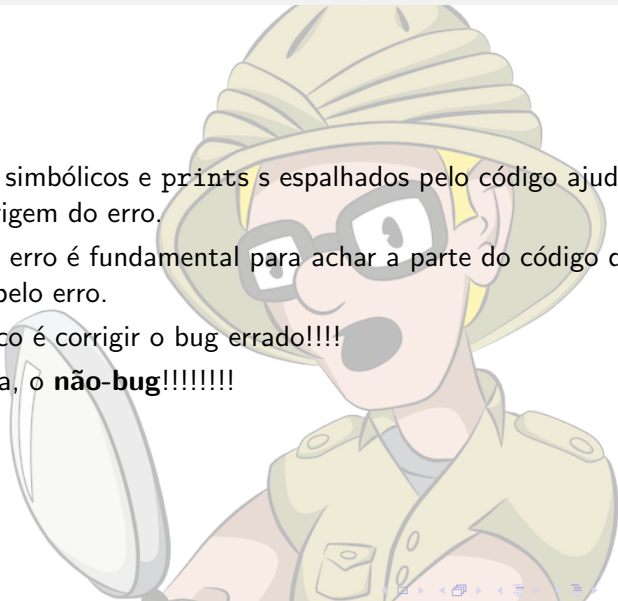
3. Localização

- Depuradores simbólicos e `prints` s espalhados pelo código ajudam a localizar a origem do erro.
- Reproduzir o erro é fundamental para achar a parte do código que é responsável pelo erro.
- O grande risco é corrigir o bug errado!!!!



3. Localização

- Depuradores simbólicos e `prints` espalhados pelo código ajudam a localizar a origem do erro.
- Reproduzir o erro é fundamental para achar a parte do código que é responsável pelo erro.
- O grande risco é corrigir o bug errado!!!!
- Ou pior ainda, o **não-bug**!!!!!!!



4. Correção



4. Correção

- Corrigir a verdadeira causa do erro.

4. Correção

- Corrigir a verdadeira causa do erro.
- Não custa nada documentar o que foi feito, principalmente se o bug for sutil.

4. Correção

- Corrigir a verdadeira causa do erro.
- Não custa nada documentar o que foi feito, principalmente se o bug for sutil.
- Muito cuidado para não introduzir novos bugs.

4. Correção

- Corrigir a verdadeira causa do erro.
- Não custa nada documentar o que foi feito, principalmente se o bug for sutil.
- Muito cuidado para não introduzir novos bugs.
- **REFAÇA** os testes, novos e velhos.

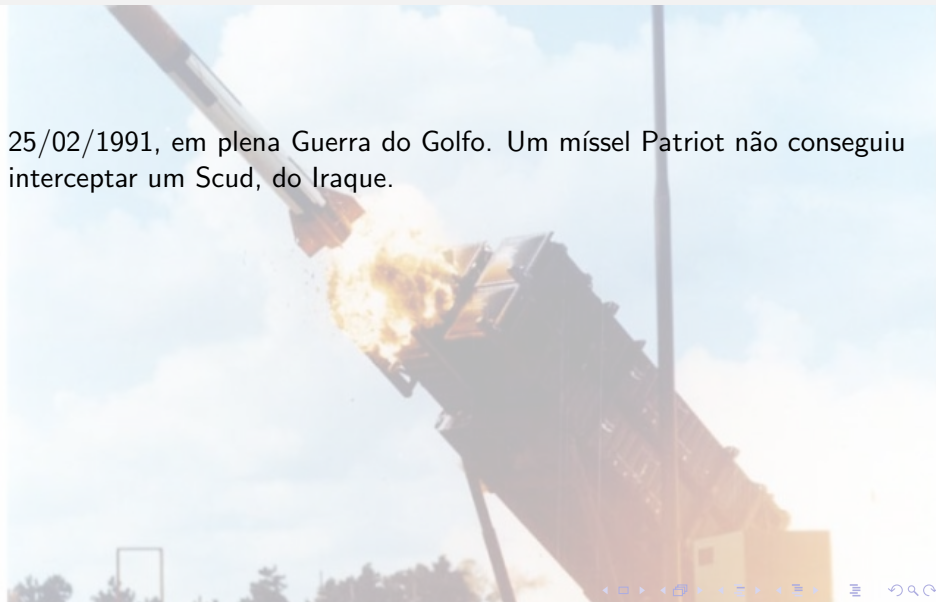
“Causos”

Alguns bugs famosos!

Casos reais que pequenos erros, mas que tiveram, ou quase tiveram, consequências significativas!!!

Patriot — 1991

25/02/1991, em plena Guerra do Golfo. Um míssil Patriot não conseguiu interceptar um Scud, do Iraque.



Patriot — 1991

25/02/1991, em plena Guerra do Golfo. Um míssil Patriot não conseguiu interceptar um Scud, do Iraque.

Origem do problema: contagem do tempo.

O relógio interno era multiplicado por $1/10$ para fornecer o tempo em segundos.

Patriot — 1991

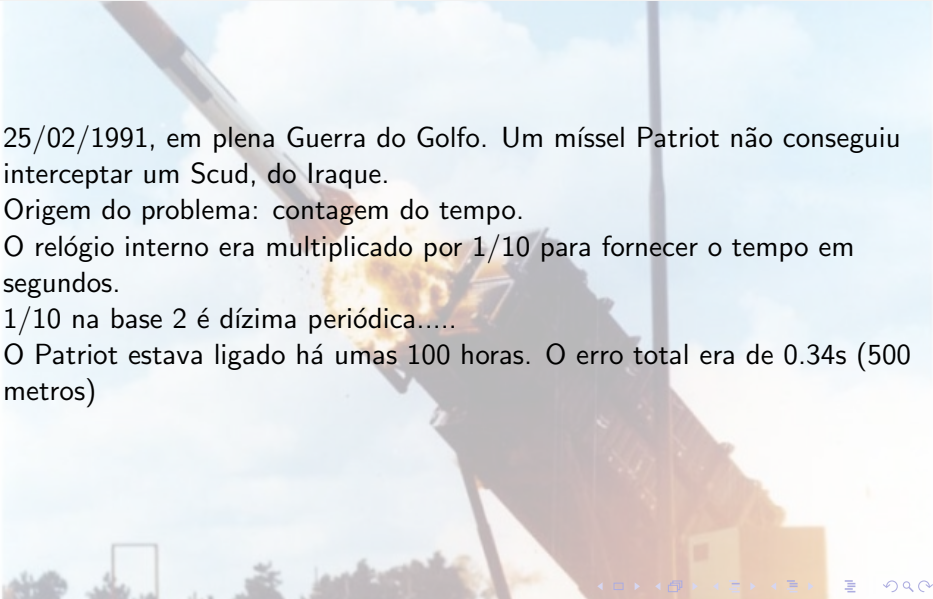
25/02/1991, em plena Guerra do Golfo. Um míssil Patriot não conseguiu interceptar um Scud, do Iraque.

Origem do problema: contagem do tempo.

O relógio interno era multiplicado por $1/10$ para fornecer o tempo em segundos.

$1/10$ na base 2 é dízima periódica.....

Patriot — 1991



25/02/1991, em plena Guerra do Golfo. Um míssil Patriot não conseguiu interceptar um Scud, do Iraque.

Origem do problema: contagem do tempo.

O relógio interno era multiplicado por $1/10$ para fornecer o tempo em segundos.

$1/10$ na base 2 é dízima periódica.....

O Patriot estava ligado há umas 100 horas. O erro total era de 0.34s (500 metros)

Patriot — 1991

25/02/1991, em plena Guerra do Golfo. Um míssil Patriot não conseguiu interceptar um Scud, do Iraque.

Origem do problema: contagem do tempo.

O relógio interno era multiplicado por $1/10$ para fornecer o tempo em segundos.

$1/10$ na base 2 é dízima periódica.....

O Patriot estava ligado há umas 100 horas. O erro total era de 0.34s (500 metros)

Custo:

Patriot — 1991

25/02/1991, em plena Guerra do Golfo. Um míssil Patriot não conseguiu interceptar um Scud, do Iraque.

Origem do problema: contagem do tempo.

O relógio interno era multiplicado por $1/10$ para fornecer o tempo em segundos.

$1/10$ na base 2 é dízima periódica.....

O Patriot estava ligado há umas 100 horas. O erro total era de 0.34s (500 metros)

Custo: 28 mortos e 100 feridos.

Ariane 5 — 1996

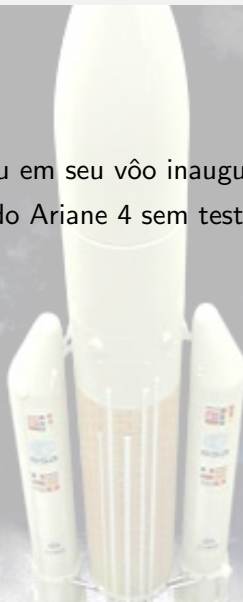
O foguete Ariane 5 explodiu em seu vôo inaugural em junho de 1996.



Ariane 5 — 1996

O foguete Ariane 5 explodiu em seu vôo inaugural em junho de 1996.

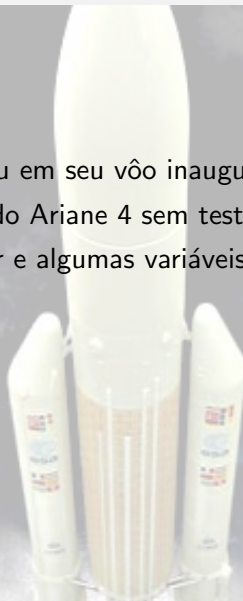
- Pacote de navegação do Ariane 4 sem teste.



Ariane 5 — 1996

O foguete Ariane 5 explodiu em seu vôo inaugural em junho de 1996.

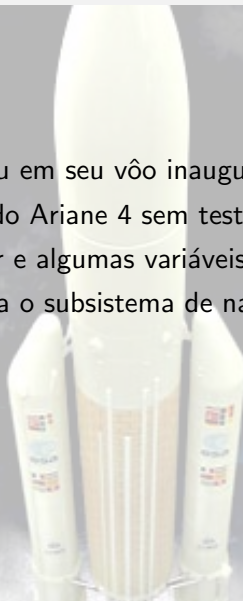
- Pacote de navegação do Ariane 4 sem teste.
- A velocidade era maior e algumas variáveis estouraram.



Ariane 5 — 1996

O foguete Ariane 5 explodiu em seu vôo inaugural em junho de 1996.

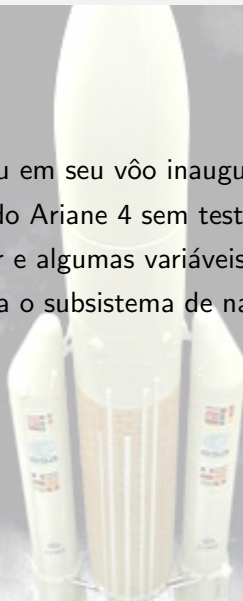
- Pacote de navegação do Ariane 4 sem teste.
- A velocidade era maior e algumas variáveis estouraram.
- O erro foi passado para o subsistema de navegação



Ariane 5 — 1996

O foguete Ariane 5 explodiu em seu vôo inaugural em junho de 1996.

- Pacote de navegação do Ariane 4 sem teste.
- A velocidade era maior e algumas variáveis estouraram.
- O erro foi passado para o subsistema de navegação

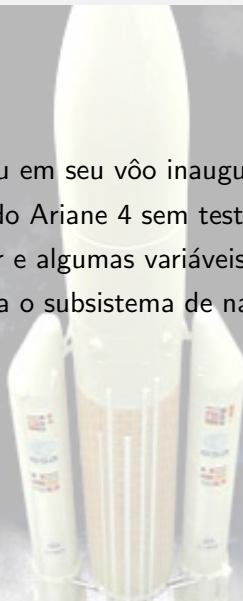


Ariane 5 — 1996

O foguete Ariane 5 explodiu em seu vôo inaugural em junho de 1996.

- Pacote de navegação do Ariane 4 sem teste.
- A velocidade era maior e algumas variáveis estouraram.
- O erro foi passado para o subsistema de navegação

Custo:

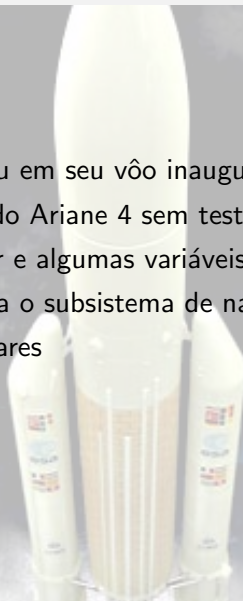


Ariane 5 — 1996

O foguete Ariane 5 explodiu em seu vôo inaugural em junho de 1996.

- Pacote de navegação do Ariane 4 sem teste.
- A velocidade era maior e algumas variáveis estouraram.
- O erro foi passado para o subsistema de navegação

Custo: 500 milhões de dólares



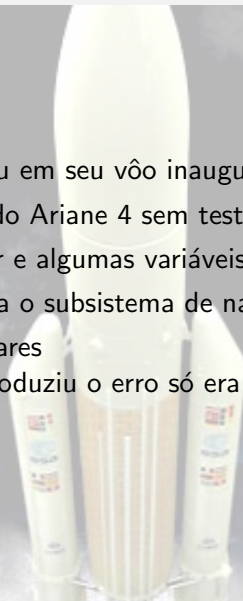
Ariane 5 — 1996

O foguete Ariane 5 explodiu em seu vôo inaugural em junho de 1996.

- Pacote de navegação do Ariane 4 sem teste.
- A velocidade era maior e algumas variáveis estouraram.
- O erro foi passado para o subsistema de navegação

Custo: 500 milhões de dólares

O pedaço do código que produziu o erro só era útil no lançamento



Mars Climate Orbiter — 1998

Depois de 286 dias de viagem até Marte a nave se aproximou muito mais do que devia.



Mars Climate Orbiter — 1998

Depois de 286 dias de viagem até Marte a nave se aproximou muito mais do que devia.

Um instrumento navegacional terciarizado calculava a saída no sistema britânico.



Mars Climate Orbiter — 1998

Depois de 286 dias de viagem até Marte a nave se aproximou muito mais do que devia.

Um instrumento navegacional terciarizado calculava a saída no sistema britânico.

A nave trabalhava no sistema métrico.

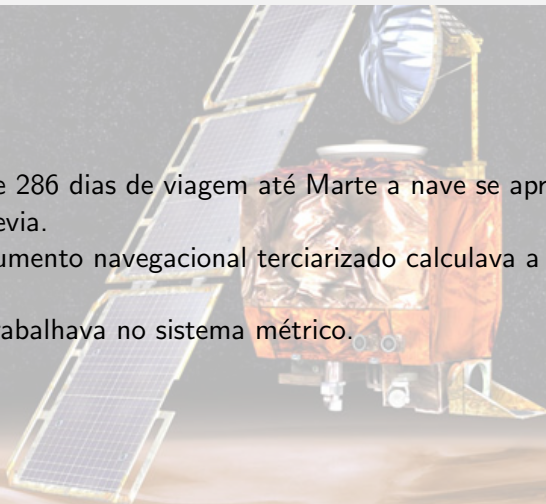
Mars Climate Orbiter — 1998

Depois de 286 dias de viagem até Marte a nave se aproximou muito mais do que devia.

Um instrumento navegacional terciarizado calculava a saída no sistema britânico.

A nave trabalhava no sistema métrico.

Custo:



Mars Climate Orbiter — 1998

Depois de 286 dias de viagem até Marte a nave se aproximou muito mais do que devia.

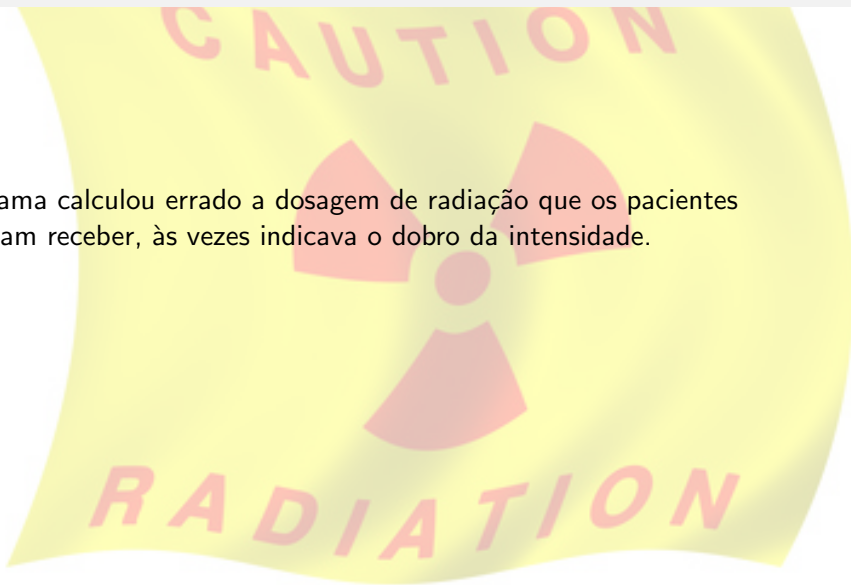
Um instrumento navegacional terciarizado calculava a saída no sistema britânico.

A nave trabalhava no sistema métrico.

Custo: 125 milhões de dólares

Tratamento de Câncer no Panamá — 2000

Programa calculou errado a dosagem de radiação que os pacientes deveriam receber, às vezes indicava o dobro da intensidade.



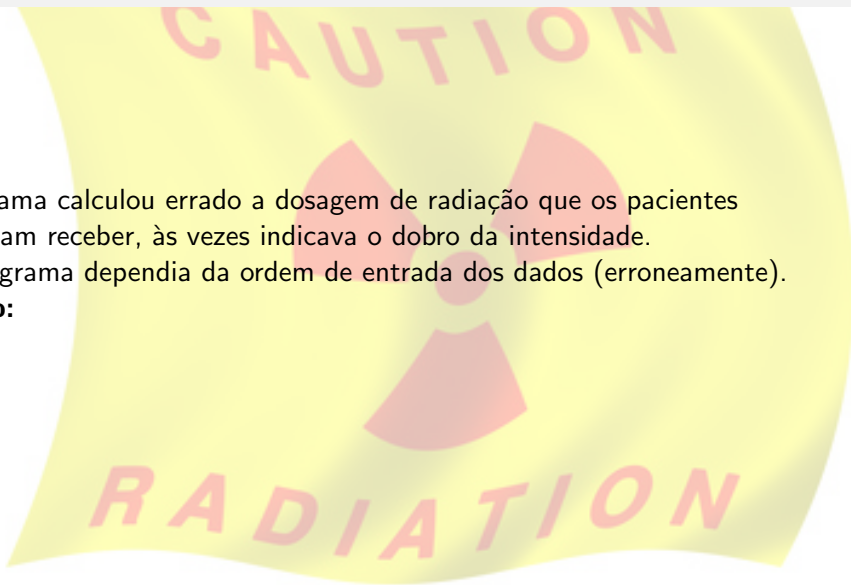
Tratamento de Câncer no Panamá — 2000

Programa calculou errado a dosagem de radiação que os pacientes deveriam receber, às vezes indicava o dobro da intensidade.
O programa dependia da ordem de entrada dos dados (erroneamente).

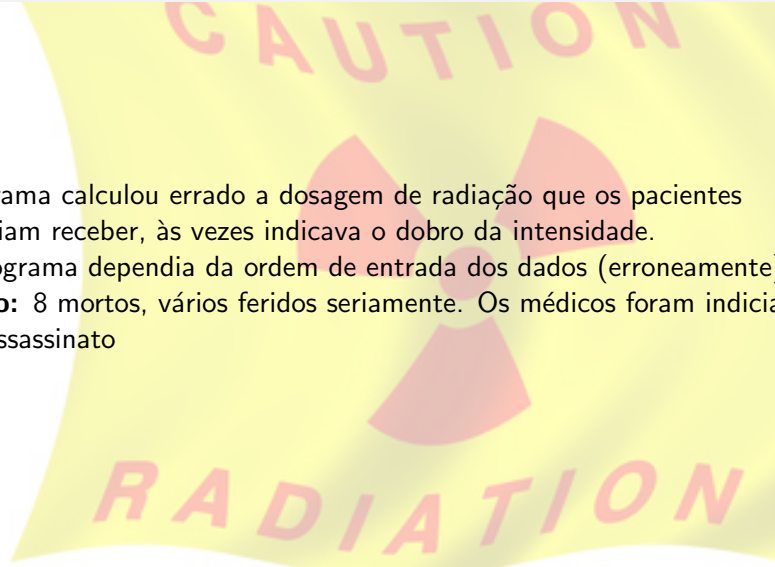
Tratamento de Câncer no Panamá — 2000

Programa calculou errado a dosagem de radiação que os pacientes deveriam receber, às vezes indicava o dobro da intensidade. O programa dependia da ordem de entrada dos dados (erroneamente).

Custo:



Tratamento de Câncer no Panamá — 2000



Programa calculou errado a dosagem de radiação que os pacientes deveriam receber, às vezes indicava o dobro da intensidade. O programa dependia da ordem de entrada dos dados (erroneamente).
Custo: 8 mortos, vários feridos seriamente. Os médicos foram indiciados por assassinato

Partido Verde na Alemanha — 1992

As eleições em Schleswig Holstein tiveram a porcentagem de delegados para nível federal maior do que deviam.

Partido Verde na Alemanha — 1992

As eleições em Schleswig Holstein tiveram a porcentagem de delegados para nível federal maior do que deviam.
Erro na apresentação: 5% ao invés de 4.97%

Partido Verde na Alemanha — 1992

As eleições em Schleswig Holstein tiveram a porcentagem de delegados para nível federal maior do que deviam.

Erro na apresentação: 5% ao invés de 4.97%

Custo:

Partido Verde na Alemanha — 1992

As eleições em Schleswig Holstein tiveram a porcentagem de delegados para nível federal maior do que deviam.

Erro na apresentação: 5% ao invés de 4.97%

Custo: poderia ter dado assentos federais aos neo-nazistas.

WW-III close call — 1993

Sistema de defesa soviético detectou 5 mísseis intercontinentais americanos.

WW-III close call — 1993

Sistema de defesa soviético detectou 5 mísseis intercontinentais americanos.

O encarregado (Stanislav Petrov) achou que 5 seria um número muito pequeno para um ataque e decidiu declarar como erro.

WW-III close call — 1993

Sistema de defesa soviético detectou 5 mísseis intercontinentais americanos.

O encarregado (Stanislav Petrov) achou que 5 seria um número muito pequeno para um ataque e decidiu declarar como erro.

O programa interpretou errado o reflexo do Sol no topo das nuvens, apesar de ter filtros especializados para isso.

WW-III close call — 1993

Sistema de defesa soviético detectou 5 mísseis intercontinentais americanos.

O encarregado (Stanislav Petrov) achou que 5 seria um número muito pequeno para um ataque e decidiu declarar como erro.

O programa interpretou errado o reflexo do Sol no topo das nuvens, apesar de ter filtros especializados para isso.

Custo:

WW-III close call — 1993

Sistema de defesa soviético detectou 5 mísseis intercontinentais americanos.

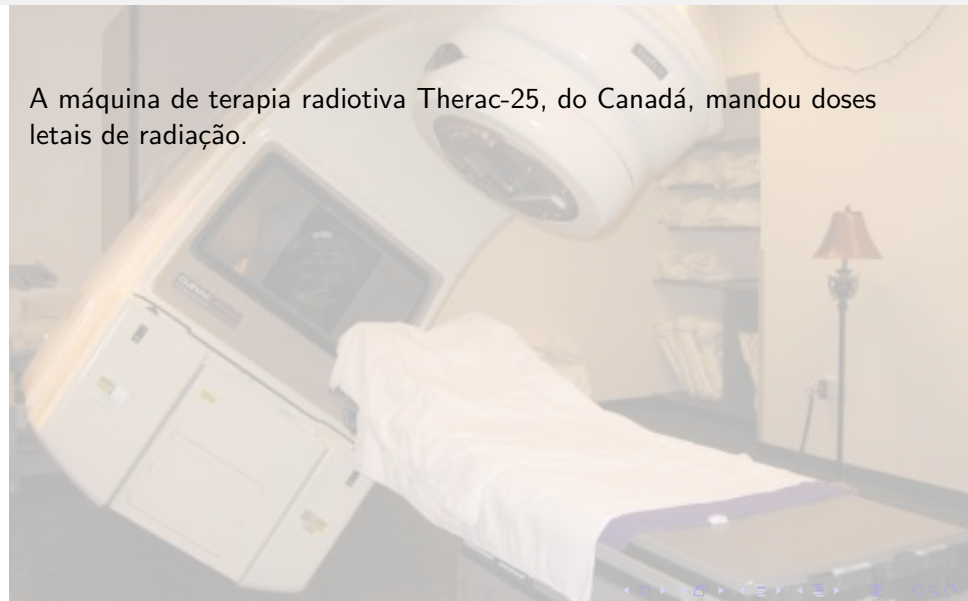
O encarregado (Stanislav Petrov) achou que 5 seria um número muito pequeno para um ataque e decidiu declarar como erro.

O programa interpretou errado o reflexo do Sol no topo das nuvens, apesar de ter filtros especializados para isso.

Custo: humanidade toda?

Therac-25 — 1985–1987

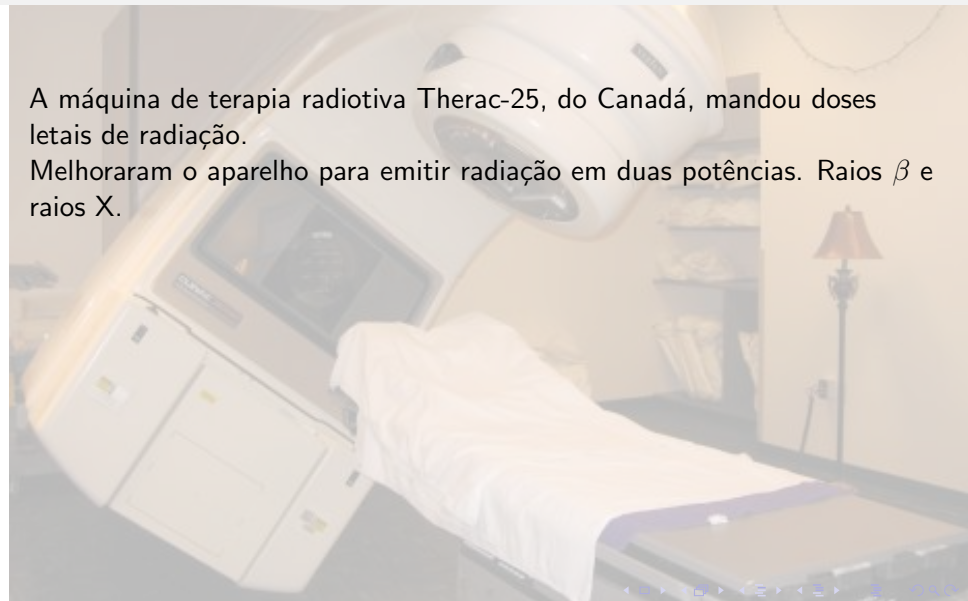
A máquina de terapia radiotiva Therac-25, do Canadá, mandou doses letais de radiação.



Therac-25 — 1985–1987

A máquina de terapia radiotiva Therac-25, do Canadá, mandou doses letais de radiação.

Melhoraram o aparelho para emitir radiação em duas potências. Raios β e raios X.



Therac-25 — 1985–1987

A máquina de terapia radiotiva Therac-25, do Canadá, mandou doses letais de radiação.

Melhoraram o aparelho para emitir radiação em duas potências. Raios β e raios X.

Havia um sistema de proteção eletromecânico, mudaram para um sistema controlado por *software*.

Therac-25 — 1985–1987

A máquina de terapia radiotiva Therac-25, do Canadá, mandou doses letais de radiação.

Melhoraram o aparelho para emitir radiação em duas potências. Raios β e raios X.

Havia um sistema de proteção eletromecânico, mudaram para um sistema controlado por *software*.

O programador era “caseiro” e não fez o controle de concorrência. Havia um problema de *race condition*.

Um digitador muito rápido ativava os dois emissores ao mesmo tempo.

Radiação β era ativada, mas com o raio X fora de posição. Por causa de uma *race condition*, a máquina disparava em momento errado.

Therac-25 — 1985–1987

A máquina de terapia radiotiva Therac-25, do Canadá, mandou doses letais de radiação.

Melhoraram o aparelho para emitir radiação em duas potências. Raios β e raios X.

Havia um sistema de proteção eletromecânico, mudaram para um sistema controlado por *software*.

O programador era “caseiro” e não fez o controle de concorrência. Havia um problema de *race condition*.

Um digitador muito rápido ativava os dois emissores ao mesmo tempo.

Radiação β era ativada, mas com o raio X fora de posição. Por causa de uma *race condition*, a máquina disparava em momento errado.

Custo:

Therac-25 — 1985–1987

A máquina de terapia radiativa Therac-25, do Canadá, mandou doses letais de radiação.

Melhoraram o aparelho para emitir radiação em duas potências. Raios β e raios X.

Havia um sistema de proteção eletromecânico, mudaram para um sistema controlado por *software*.

O programador era “caseiro” e não fez o controle de concorrência. Havia um problema de *race condition*.

Um digitador muito rápido ativava os dois emissores ao mesmo tempo.

Radiação β era ativada, mas com o raio X fora de posição. Por causa de uma *race condition*, a máquina disparava em momento errado.

Custo: 5 mortos e vários afetados seriamente

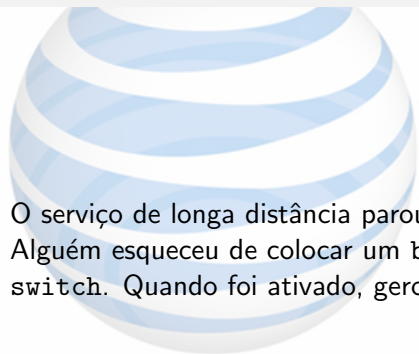
AT&T — 1990



at&t

O serviço de longa distância parou por 9 horas.

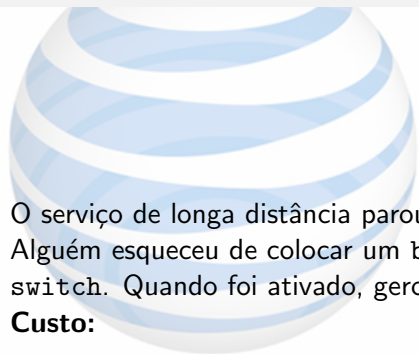
AT&T — 1990



at&t

O serviço de longa distância parou por 9 horas.
Alguém esqueceu de colocar um break no final de um case de um switch. Quando foi ativado, gerou um efeito cascata.

AT&T — 1990

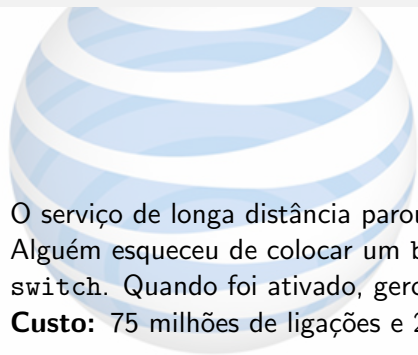


at&t

O serviço de longa distância parou por 9 horas.
Alguém esqueceu de colocar um break no final de um case de um switch. Quando foi ativado, gerou um efeito cascata.

Custo:

AT&T — 1990



at&t

O serviço de longa distância parou por 9 horas.
Alguém esqueceu de colocar um break no final de um case de um switch. Quando foi ativado, gerou um efeito cascata.

Custo: 75 milhões de ligações e 200000 reservas aéreas perdidas.

Outras curiosidades

- Ano 2000



Outras curiosidades

- Ano 2000
- Torpedos que evitam retorno



Outras curiosidades

- Ano 2000
- Torpedos que evitam retorno
- Caça cruzando o Equador



Outras curiosidades

- Ano 2000
- Torpedos que evitam retorno
- Caça cruzando o Equador
- Erro do FORTRAN (1962)



Outras curiosidades

- Ano 2000
- Torpedos que evitam retorno
- Caça cruzando o Equador
- Erro do FORTRAN (1962)
- Gemini V errou o ponto de aterrissagem porque o programa não levava em conta o movimento de translação.

Outras curiosidades

- Ano 2000
- Torpedos que evitam retorno
- Caça cruzando o Equador
- Erro do FORTRAN (1962)
- Gemini V errou o ponto de aterrissagem porque o programa não levava em conta o movimento de translação.
- IBM 7094 no MIT

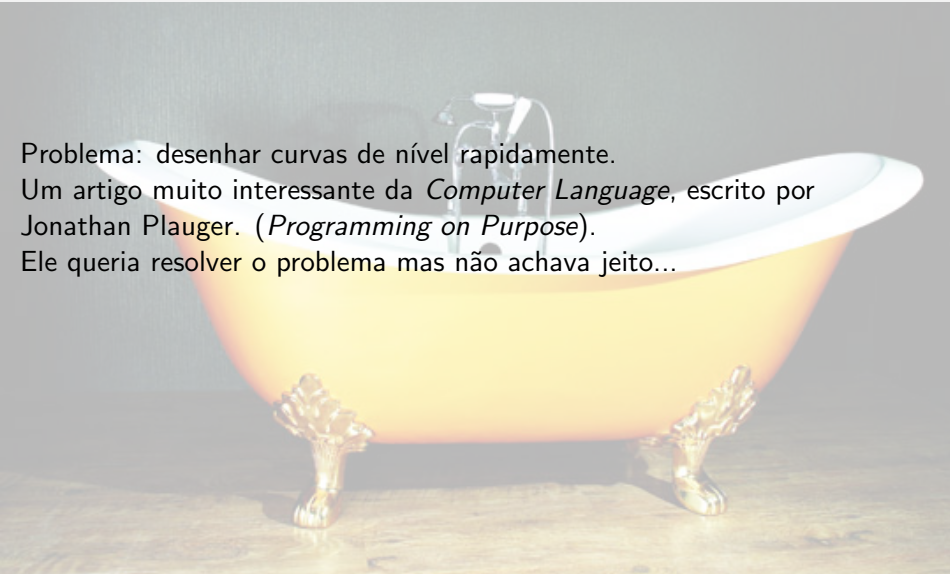
Outras curiosidades

- Ano 2000
- Torpedos que evitam retorno
- Caça cruzando o Equador
- Erro do FORTRAN (1962)
- Gemini V errou o ponto de aterrissagem porque o programa não levava em conta o movimento de translação.
- IBM 7094 no MIT
 - Um processo de baixa prioridade estava esperando na fila desde 1967.

Outras curiosidades

- Ano 2000
- Torpedos que evitam retorno
- Caça cruzando o Equador
- Erro do FORTRAN (1962)
- Gemini V errou o ponto de aterrissagem porque o programa não levava em conta o movimento de translação.
- IBM 7094 no MIT
 - Um processo de baixa prioridade estava esperando na fila desde 1967.
 - Só foi descoberto quando o computador ia ser desligado em 1973

O algoritmo da banheira



Problema: desenhar curvas de nível rapidamente.
Um artigo muito interessante da *Computer Language*, escrito por Jonathan Plauger. (*Programming on Purpose*).
Ele queria resolver o problema mas não achava jeito...

O algoritmo da banheira

Problema: desenhar curvas de nível rapidamente.

Um artigo muito interessante da *Computer Language*, escrito por Jonathan Plauger. (*Programming on Purpose*).

Ele queria resolver o problema mas não achava jeito...

Resolveu tomar um banho...

O algoritmo da banheira

Problema: desenhar curvas de nível rapidamente.

Um artigo muito interessante da *Computer Language*, escrito por Jonathan Plauger. (*Programming on Purpose*).

Ele queria resolver o problema mas não achava jeito...

Resolveu tomar um banho...

A água na banheira desenhava automaticamente as curvas de nível em seu corpo!

Como escrever?

O algoritmo para $F(x, y)$

Discretize o retângulo.

Para cada ponto (x, y) :

Se $F(x, y)$ for múltiplo de H

Desenha um ponto preto

Senão

Desenha um ponto branco

Fim

O algoritmo para $F(x, y)$

Discretize o retângulo.

Para cada ponto (x, y) :


Se $F(x, y)$ for múltiplo de H (a menos de um erro pequeno)

Desenha um ponto preto

Senão

Desenha um ponto branco

Fim

A yellow clawfoot bathtub with chrome fixtures, including a showerhead and faucet, set against a grey background. The text "Implementação e resultado...." is overlaid on the tub.

Implementação e resultado....

Explicação

Ele resolveu tomar outro banho

Explicação

Ele resolveu tomar outro banho
mas desta vez de banheira, foi para um chuveiro

Explicação

Ele resolveu tomar outro banho
mas desta vez de banheira, foi para um chuveiro
As gotas caindo no chão formam um padrão de interferência...

Explicação

Ele resolveu tomar outro banho
mas desta vez de banheira, foi para um chuveiro
As gotas caindo no chão formam um padrão de interferência...
Muito parecido com o resultado: onde é gerada a interferência?

Explicação

Ele resolveu tomar outro banho
mas desta vez de banheira, foi para um chuveiro
As gotas caindo no chão formam um padrão de interferência...
Muito parecido com o resultado: onde é gerada a interferência?
Duas ondas:

- Uma amostragem a intervalos regulares no plano XY

Explicação

Ele resolveu tomar outro banho
mas desta vez de banheira, foi para um chuveiro
As gotas caindo no chão formam um padrão de interferência...
Muito parecido com o resultado: onde é gerada a interferência?
Duas ondas:

- Uma amostragem a intervalos regulares no plano XY
- Uma amostragem a alturas regulares (múltiplos de H), “refletidas na superfície”

Explicação

Ele resolveu tomar outro banho
mas desta vez de banheira, foi para um chuveiro
As gotas caindo no chão formam um padrão de interferência...
Muito parecido com o resultado: onde é gerada a interferência?
Duas ondas:

- Uma amostragem a intervalos regulares no plano XY
- Uma amostragem a alturas regulares (múltiplos de H), “refletidas na superfície”

Explicação

Ele resolveu tomar outro banho
mas desta vez de banheira, foi para um chuveiro
As gotas caindo no chão formam um padrão de interferência...
Muito parecido com o resultado: onde é gerada a interferência?
Duas ondas:

- Uma amostragem a intervalos regulares no plano XY
- Uma amostragem a alturas regulares (múltiplos de H), “refletidas na superfície”

Acabamos por contruir um holograma da superfície!!!



That's all Folks!

That's all Folks!

